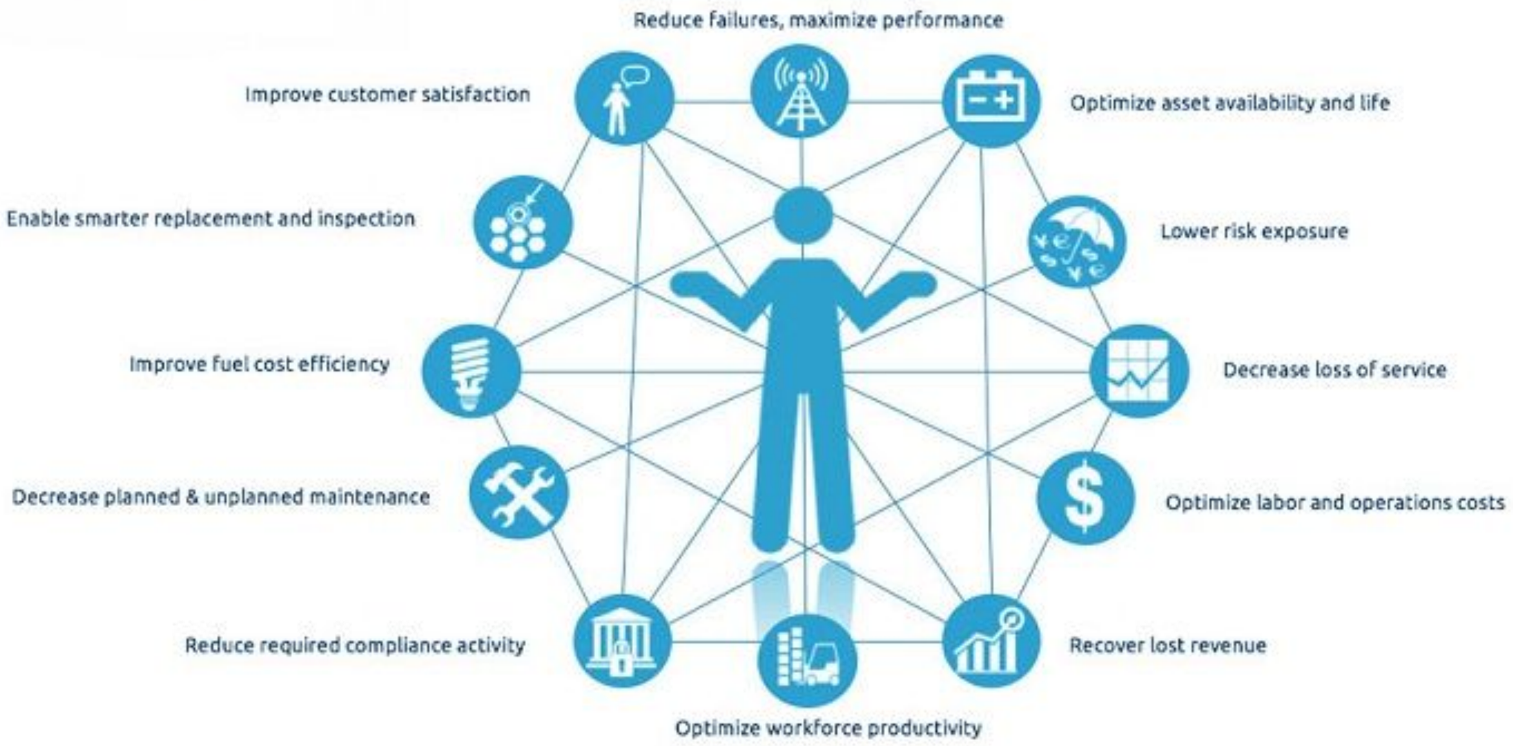
Equipment Failure Prediction

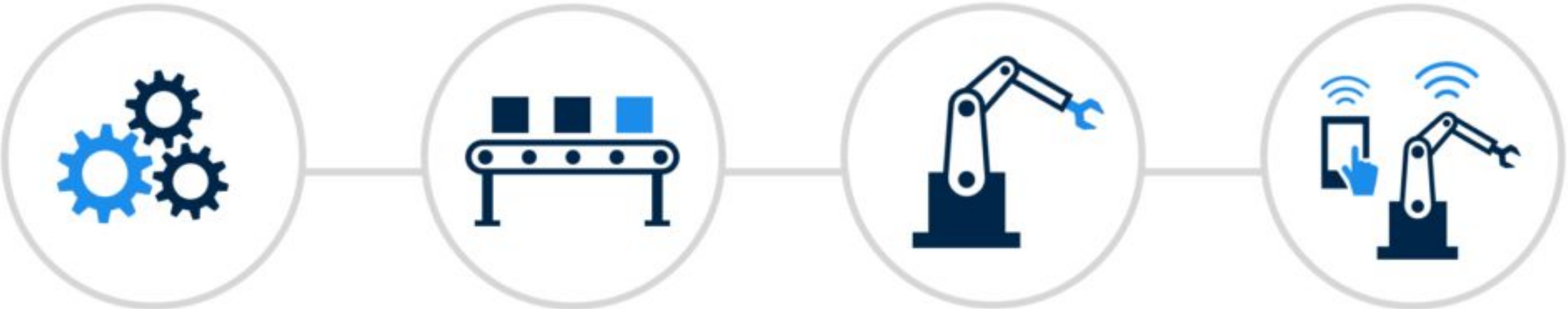
TEAM - CODE CRACKERS
Vidhi Kansara (**Team Lead**)
Rohit Roshan
Suresh Narayanan
Anjaneyulu Sunkara

MENTORED BY NAHAS PAREEKUTTY

Why Predictive maintenance ?



The Four Industrial Revolutions



Industry 1.0

Mechanization and the introduction of steam and water power

Industry 2.0

Mass production assembly lines using electrical power

Industry 3.0

Automated production, computers, IT-systems and robotics

Industry 4.0

The Smart Factory. Autonomous systems, IoT, machine learning

Growth potential

- The global predictive maintenance market size is projected to reach [USD 18,551.0] million by 2026

- Increasing number of company mergers has made a huge impact on market growth.

- The CXP Group report, *Digital Industrial Revolution with Predictive Maintenance*, revealed that 91 percent of predictive maintenance manufacturers' see a reduction of repair time and unplanned downtime, and 93 percent see improvement of aging industrial infrastructure.

- According to a PWC report, predictive maintenance in factories could:

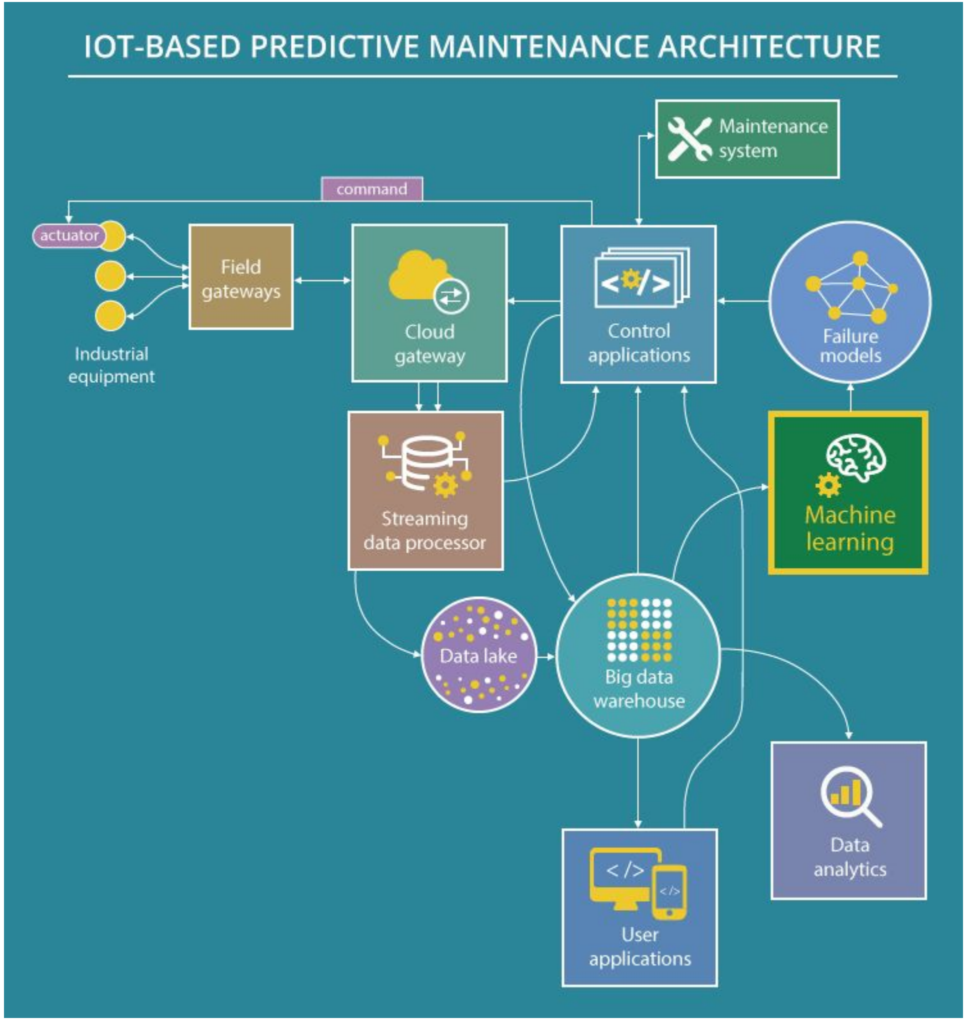
    1. Reduce cost by 12 percent
    2. Improve uptime by 9 percent
    3. Reduce safety, health, environment, and quality risks by 14 percent
    4. Extend the lifetime of an aging asset by 20 percent

HOW IT WORKS

Various types of sensors & equipment are used to evaluate an asset's performance in real-time. IoT allows for different assets and systems to connect, work together, and share, analyze and action data.

The data collected previously is **analyzed using predictive algorithms** that identify trends to detect when an asset will require repair, servicing, or replacement.

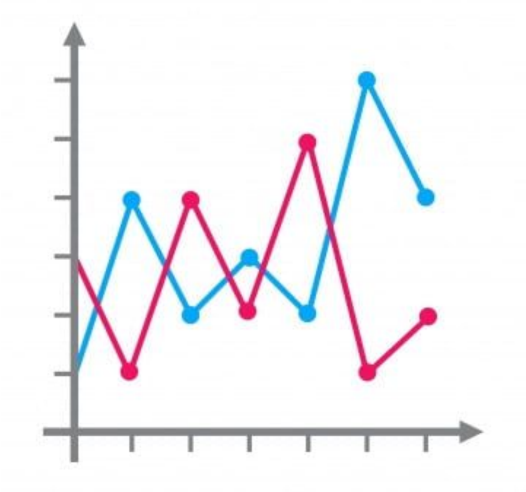
These algorithms follow a set of predetermined rules that compare the asset's current behavior against its expected behavior. **Deviations are an indication of gradual deterioration** that will lead to asset failure.



Source : Google Images

WORKFLOW

STEP 1



Fetched Industrial Machine Failure Dataset having sufficient data & features

STEP 2

Data was cleaned & analysed using Python modules such as Pandas, Numpy

STEP 3

Relevant machine learning method were selected and model was built, trained & tested

TOOLS USED for Implementation

python™

pandas

scikits learn

NumPy

TensorFlow

K Keras

G colab

STEP 1 : DATASET Collection

PARAMETERS CONSIDERED FOR SEARCHING A DATASET

1. Sufficient number of features i.e more than 10
2. Sufficient dataset size i.e index more than 8000
3. Industry relevant dataset
4. Actual machine readings of particular industry.

DATASET FROM
kaggle.com

FINAL DATASET SPECIFICATION SELECTED FOR FURTHER ANALYSIS

1. Number of columns/features = 36
2. Training dataset size/rows = 20867, Test dataset size/rows = 7089
3. The dataset has telemetry reading and error identification, maintenance, and failure:
4. Industrial Motor - Volt, telemetry, pressure,errors, and vibration are measured for a period of 24 hrs and 5 days

DATASET :
https://www.kaggle.com/tiagotgoz/predictive-useful-life-based-into-telemetry?select=ALLtrainMescla5D.csv

STEP 2: Data Analysis

Max life cycle for each machine ID was calculated

	machineID	MaxCycleID
0	1	346
1	2	360
2	3	333
3	4	283
4	5	275
...	...	...
93	96	239
94	97	321
95	98	328
96	99	329
97	100	339

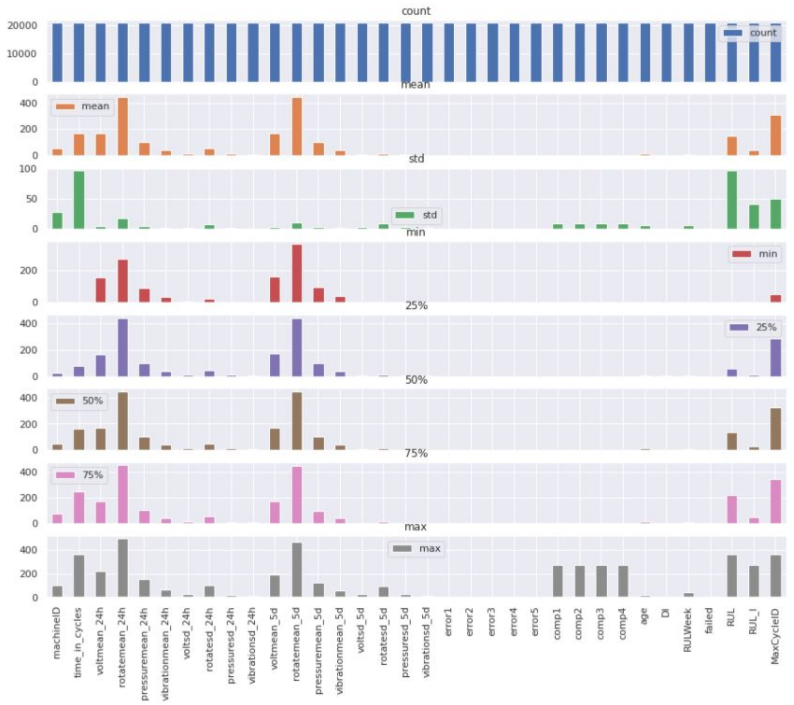
98 rows × 2 columns

	time_in_cycles	RUL
0	2	344
1	3	343
2	4	342
3	5	341
4	6	340
...	...	...
20862	335	4
20863	336	3
20864	337	2
20865	338	1
20866	339	0

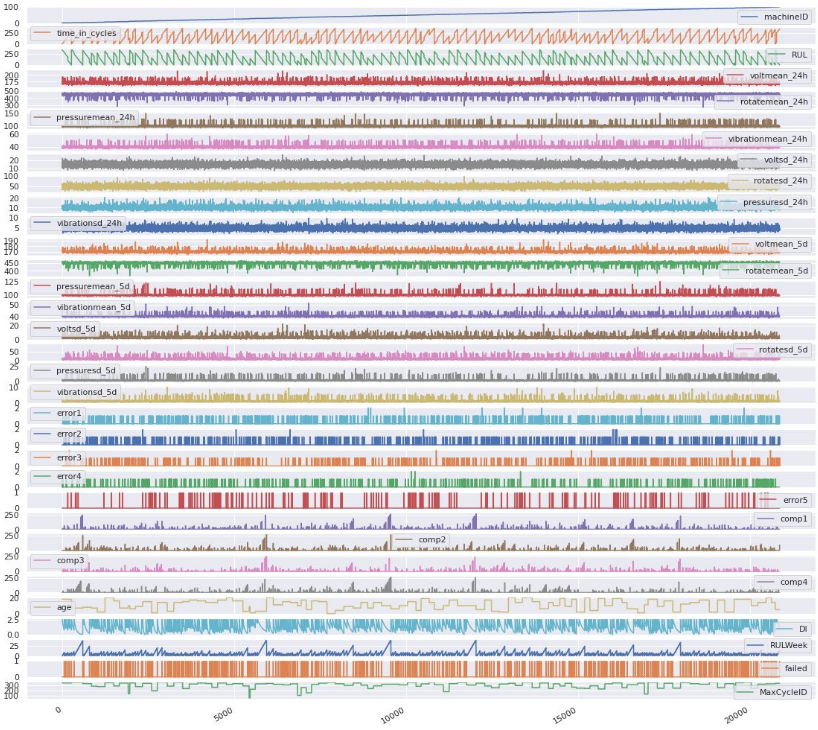
20867 rows × 2 columns

Remaining useful life = MaxcycleID - time_in_cycles was calculated

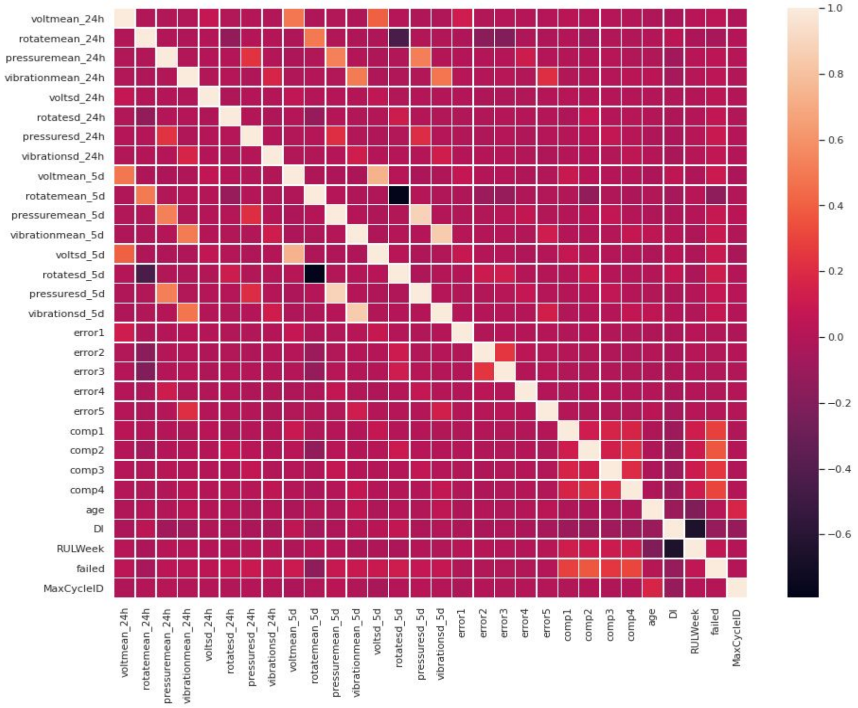
After merging RUL and MaxCycleID to the dataset, its statistical specification were plotted i.e count, mean, standard deviation, minimum, maximum, and 25%,50%,75% of maximum



Data Analysis Contd.



All the sensor values were plotted in time domain to get an intuition of the data



The correlation matrix for the dataset was plotted to find highly correlated values and discard features which do not affect any other features

Preprocessing of Data for prediction

Fail_lable variable was chosen to specify if the machine's **Remaining Useful Life (RUL)** is less than a given period of time in days, months, years.

The **period was set to 30** i.e if RUL < 30 days then **Fail_label will be set to 1** which signifies that machines life is less than 30 days on that particular Data.

```
Train_Data_copy = Train_Data.copy()
Period = 30
Train_Data_copy['fail_label'] = Train_Data_copy['RUL'].apply(lambda x: 1 if x <= Period else 0)
Train_Data_copy.head(10)
```

RUL	fail_label
344	0
343	0
342	0
341	0
340	0
339	0
338	0
337	0
336	0
335	0

RUL = MAX OPERATED CYCLE - CURRENT CYCLE
= MaxCycleID - Time_in_cycles

If RUL< = 30, then Fail_label = 1
Else if RUL>30, then Fail_label = 0

Feature selection & Model Summary

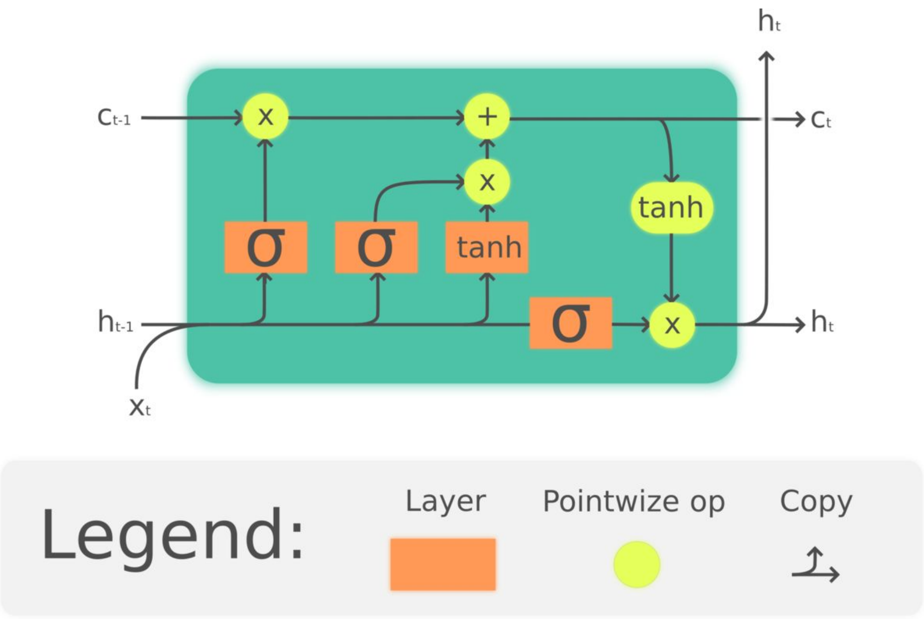
```
feature_cols = ['voltmean_24h','rotatemean_24h','pressuremean_24h','vibrationmean_24h',
                'voltsd_24h','rotatesd_24h',
                'pressuresd_24h','vibrationsd_24h','voltmean_5d','rotatemean_5d',
                'pressuremean_5d','vibrationmean_5d','voltsd_5d','rotatesd_5d',
                'pressuresd_5d','vibrationsd_5d','age','DI']

Target_cols = ['fail_label']
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 50, 50)            13800
_________________________________________________________________
dropout (Dropout)            (None, 50, 50)            0
_________________________________________________________________
lstm_1 (LSTM)                (None, 25)                7600
_________________________________________________________________
dropout_1 (Dropout)          (None, 25)                0
_________________________________________________________________
dense (Dense)                (None, 1)                 26
=================================================================
Total params: 21,426
Trainable params: 21,426
Non-trainable params: 0
_________________________________________________________________
```

Features such as Errors,components, error type were dropped as they were either constant throughout or had string values which hardly affects the outcome.

A sequential model, long short term memory (LSTM) model built around the analysed data

LSTM Model



Source : Google Images

WHY LSTM?

Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies are used to predict failures.

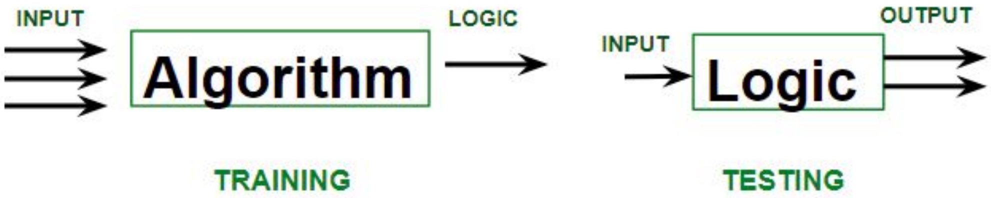
LSTM help to identify the changes in the machine condition over time.

As our data was sequential reading of sensor values of a particular machineID , LSTM was the preferred fit.

LSTM will check the past/future values as well as present values to predict outputs which will be a beneficial choice in this case to achieve a flexible and accurate model

Step 3: Model Training

The model was trained by tuning model parameters to fit on the training dataset properly and to avoid over and under fitting situations the training was monitored closely



DURING TRAINING OF OUR MODEL

```
Epoch 1/10
416/416 [==============================] - 11s
26ms/step - loss: 0.4074 - accuracy: 0.8514 -
val_loss: 0.3984 - val_accuracy: 0.8519
Epoch 2/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3979 - accuracy: 0.8543 -
val_loss: 0.3960 - val_accuracy: 0.8519
Epoch 3/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3963 - accuracy: 0.8543 -
val_loss: 0.3985 - val_accuracy: 0.8519
Epoch 4/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3947 - accuracy: 0.8543 -
val_loss: 0.3985 - val_accuracy: 0.8519
Epoch 5/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3942 - accuracy: 0.8543 -
val_loss: 0.3952 - val_accuracy: 0.8519
Epoch 6/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3916 - accuracy: 0.8543 -
val_loss: 0.3958 - val_accuracy: 0.8519
Epoch 7/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3930 - accuracy: 0.8543 -
val_loss: 0.3951 - val_accuracy: 0.8519
Epoch 8/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3931 - accuracy: 0.8543 -
val_loss: 0.3949 - val_accuracy: 0.8519
Epoch 9/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3931 - accuracy: 0.8543 -
val_loss: 0.3991 - val_accuracy: 0.8519
Epoch 10/10
416/416 [==============================] - 10s
24ms/step - loss: 0.3916 - accuracy: 0.8543 -
val_loss: 0.3953 - val_accuracy: 0.8519
```

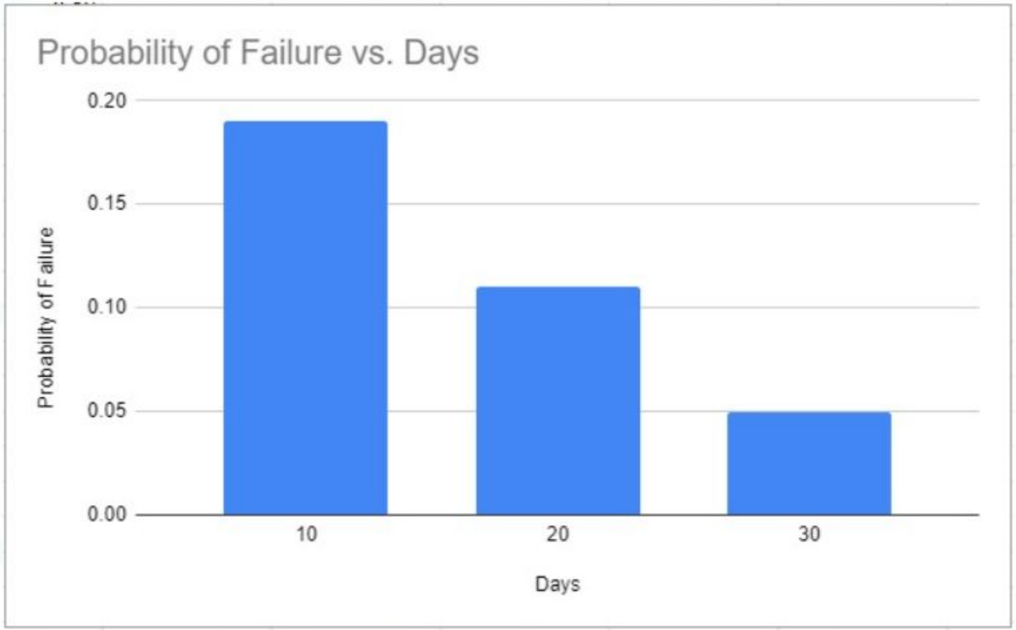
Training and Validation Analysis

Our Model achieved an accuracy of 85%

- Accuracy can be improved with a larger training set and more precise selection of features and using technologies like cloud, IoT, Big Data to name a few.
- More accurate model can be built through proper mathematical and scientific analysis of dataset
- Development of new ML Model trained and tested for specific industry e.g Aerospace, Manufacturing etc

OUTCOME:

OUR MODEL WAS SUCCESSFUL IN PREDICTING THE MACHINE FAILURE PROBABILITY IN ADVANCE



X AXIS - DAYS LEFT

Y AXIS - PROBABILITY OF FAILURE

THE FOLLOWING GRAPH SHOWS THE VARIATION OF FAILURE PROBABILITY WHICH INCREASES AS THE MACHINE REACHES ITS FINAL WORKING CYCLE

Current gaps in PdM

- Inadequate detailed technical knowledge about the equipment.
- Selection of appropriate devices for data acquisition.
- The need for proven analytics and mathematical models to convert data acquired into insights.
- Down time involved in retrofitting and monitoring assets before making them available for production.
- predictive maintenance has a high upfront cost.

Infosys in PdM

- Internet of Things (IoT) – to monitor critical components
- Real-time data analytics – to receive updates on data variations and Implications
- Algorithmic prediction models – for insights into potential failure situations & Machine learning – to receive alerts on Risk
- Augmented reality, virtual reality,and mixed reality technologies – to troubleshoot and inform on problems so that engineers are prepared to
  Respond at destination.

Future of PdM

Leveraging predictive maintenance with a high level of precision, manufacturers can focus on differentiating products using digital capabilities like self-awareness of technical health.

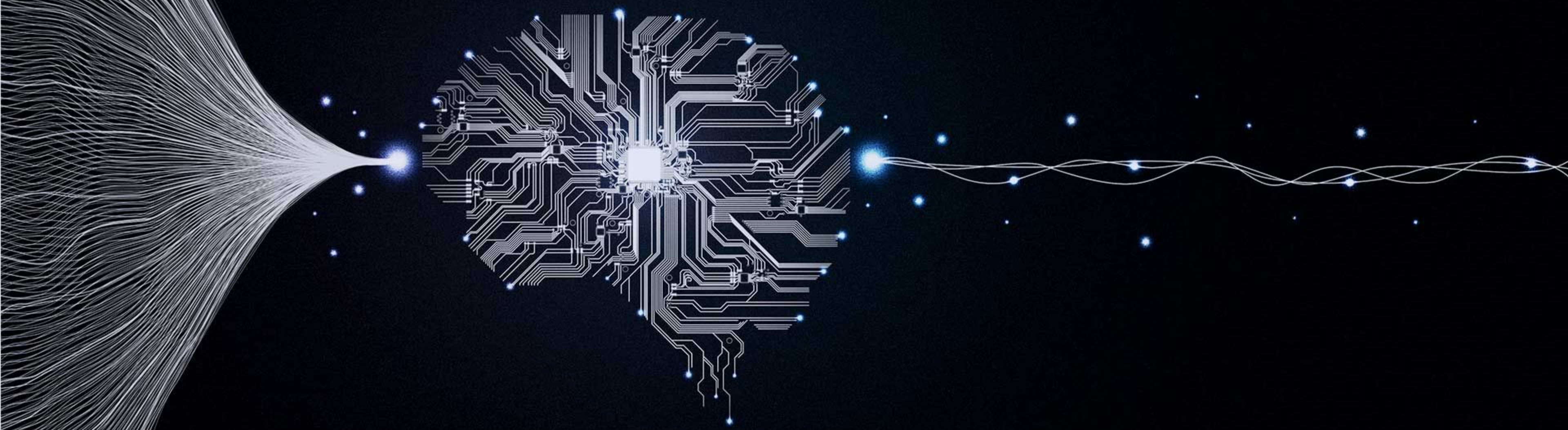
whole new level of production **efficiency** can be achieved.

In a mission-critical situation, a prescriptive system will autonomously decide what to do. This is how the predictive maintenance with IoT would drive **industry 4.0 revolution.**

References

1. https://www.globenewswire.com/news-release/2020/08/31/208
42/0/en/Predictive-Maintenance-Market-to-Rise-at-29-8-CAGR-t
2026-Increasing-Number-of-Company-Collaborations-to-Aid-Gr
h-says-Fortune-Business-Insights.html
2. https://www.anodot.com/blog/predictive-maintenance/
3. https://www.kaggle.com/tiagotgoz/predictive-useful-life-based-i
o-telemetry?select=ALLtrainMescla5D.csv
4. https://colah.github.io/posts/2015-08-Understanding-LSTMs/
5. https://www.prometheusgroup.com/posts/the-present-and-futu
of-predictive-maintenance-and-reliability-a-practitioners-perspe
e
6. https://www.google.com/search?q=predictive+maintenance+fut
&tbm=isch&ved=2ahUKEwiB-4DkoZPsAhUtnUsFHRIIAEMQ2-c
QIABAA&oq=predictive+maintenance+future&gs_lcp=CgNpbW
AzIECAAQGDoECAAQQzoCCAA6BggAEAgQHlDSPFiQUWDaU
BcAB4AoABggOIAaoKkgEHMy42LjAuMZgBAKABAaoBC2d3cy
aXotaW1nwAEB&sclient=img&ei=8bl1X8G-Cq26rtoPmZCBmAQ
h=722&biw=1536&rlz=1C1CHBF_enIN914IN914#imgrc=FpRQmE
--OokM
7. https://machinelearningmastery.com/time-series-prediction-lstn
current-neural-networks-python-keras/
8. https://towardsdatascience.com/illustrated-guide-to-lstms-and-
-s-a-step-by-step-explanation-44e9eb85bf21
9. Google images , Vector images, Backgrounds neural network, logos

THANK YOU



CHECK OUT OUR PROJECT AT GOOGLE COLAB LINK

https://colab.research.google.com/drive/1i3NGIgTwMXxBmCfsVkmDfBnjatF3ifRT?authuser=1